capC-MAP: software for analysis of Capture-C data

Genome analysis

capC-MAP: software for analysis of Capture-C data

Adam Buckle [2], Nick Gilbert [2], Davide Marenduzzo [1] and Chris A. Brackley [1,*]

[1]SUPA, School of Physics and Astronomy, University of Edinburgh, Peter Guthrie Tait Road, Edinburgh EH9 3FD, United Kingdom and [2]Medical Research Council Human Genetics Unit, Medical Research Council Institute of Genetics & Molecular Medicine, University of Edinburgh, Western General Hospital, Edinburgh, United Kingdom.

*To whom correspondence should be addressed.

Abstract

Summary: Capture-C is a member of the chromosome-conformation-capture family of experimental methods which probes the 3-D organisation of chromosomes within the cell nucleus. It provides high-resolution information on the genome-wide chromatin interactions from a set of "target" genomic locations, and is growing in popularity as a tool for improving our understanding of *cis*-regulation and gene function. Yet, analysis of the data is complicated, and to date there has been no dedicated or easy-to-use software to automate the process. We present capC-MAP, a software package for the analysis of Capture-C data.

Availability and Implementation: Implemented with both ease of use and flexibility in mind, capC-MAP is a suit of programs written in C++ and Python, where each program can be run separately, or an entire analysis can be performed with a single command line. It is available under an open-source licence at https://github.com/cbrackley/capC-MAP, as well as via the conda package manager, and should run on any standard Unix-style system.

Contact: C.Brackley@ed.ac.uk

Supplementary information: Supplementary data are available at *Bioinformatics* online.

This is a pre-copyedited, author-produced version of an article accepted for publication in Bioinformatics following peer review. The version of record is available online at: https://academic.oup.com/bioinformatics doi: 10.1093/bioinformatics/btz480.

1 Introduction

Over recent years the family of experimental methods based on chromosome-conformation-capture (3C) has grown (Han *et al.*, 2018), with different variants used to generate data at different resolutions, using different methods of detection – e.g. PCR, microarray, or next-generation sequencing (NGS) technologies. These methods are used to probe the interactions between different chromatin regions *in vivo*, and to uncover the three-dimensional organization of chromosomes and genomes. In the near two decades since they were first developed, they have revolutionised our understanding of genome organisation and function (Denker and de Laat, 2016).

The 3C based protocols range from the original "one-to-one" 3C method (Dekker *et al.*, 2002) which measures interactions between selected pairs of genomic loci; through the "one-to-all" style 4C method (Simonis *et al.*, 2006), where genome-wide interactions for a single selected locus are obtained; to high-throughput "all-to-all" HiC (Lieberman-Aiden *et al.*, 2009), which uses NGS to obtain genome-wide chromatin interaction maps. Capture-C (or NG Capture-C) is a relatively recent addition to the 3C family, developed by Hughes *et al.* (2014); it uses oligo-capture technologies, a frequently cutting restriction enzyme, and NGS sequencing, to deliver high-resolution *cis*-interaction profiles for up to hundreds of target loci from a single experiment. While HiC can provide a large-scale overview of chromosome interactions, deep sequencing is required to get good spatial resolution, which is costly. Capture-C is a "many-to-all" assay which gives interaction profiles for a set of "targets" at near restriction enzyme fragment resolution (Hughes *et al.*, 2014; Davies *et al.*, 2016).

The popularity of the Capture-C method has grown (Andrey *et al.*, 2017; Furlan *et al.*, 2018; Buckle *et al.*, 2018), but the analysis of the data is complicated – it requires non-standard use of bioinformatics tools as well as some bespoke data treatment. Most work using the method to date has used custom analysis scripts accessible only to experts in bioinformatics and programming. While some analysis tools designed to treat HiC data now also support Capture-C, these are not optimized for the method and are limited in functionality: there has been a lack of easy-to-use, dedicated software. Here we introduce capC-MAP, a software package for the analysis of Capture-C data.

2 capC-MAP

capC-MAP is implemented as a suit of programs written in C++ and Python. It calls several several common external software packages (cutadapt (Martin, 2011), bowtie (Langmead *et al.*, 2009) and samtools (Li *et al.*, 2009)), as well as performing Capture-C specific processing steps (namely, "target" and "reporter" restriction enzyme fragments are identified, invalid interactions are removed, and pile-ups of interactions are generated for each target). Full details of these processing steps are given in Supplementary Information.

An entire Capture-C analysis can be performed with a single command, or, for bespoke analyses, the capC-MAP component programs can be run separately. Usage of the software is docu-

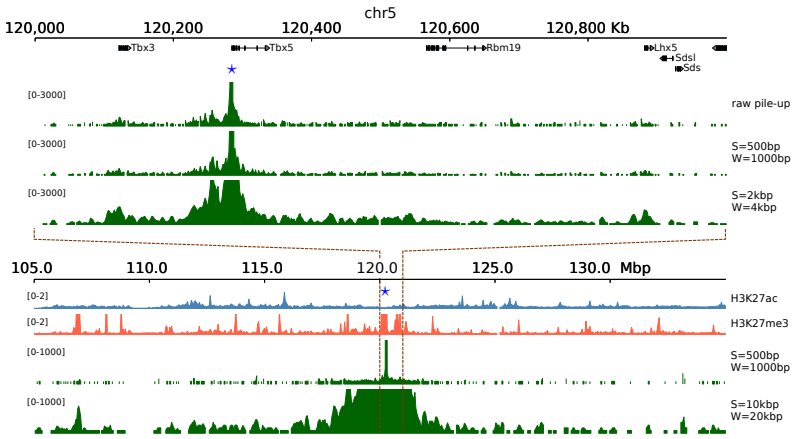


Fig. 1: Typical plots from a Capture-C experiment. Data from Andrey *et al.* (2017) (GEO:GSM2251422) were used to generate plots showing interactions of the mouse *Tbx5* gene promoter in mouse embryonic (E10.5) midbrain cells. capC-MAP generates raw pile-ups of interactions as well as binned and smoothed interaction profiles (a sliding window of length W is used to generate bins of width S). Top: interaction profiles with different bin widths. Bottom: interactions from a wider region are shown alongside ChIP-seq data for histone modifications (from the same reference). Different bin widths reveal interactions at different scales. The star indicates the position of the target.

mented in its user manual, but a typical work flow is as follows. First, the user builds an index for the reference genome to which the data will be aligned; then capC-MAP is used to generate a restriction enzyme fragment map from the same reference. These steps only need to be performed once for a given genome. Finally, the capC-MAP pipe-line is run on the input data (a pair of fastq files obtained from paired-end sequencing). The main output is an "interaction profile" (reads *vs.* genome position) for each target locus (Fig. **1**). This signal is proportional to the number of cells within the population where the target was spatially proximate to the given genomic position. capC-MAP can perform normalization to remove biases arising because different oligos have different capture efficiencies, and different binning and smoothing options can be applied to reveal features at different length scales (compare tracks in Fig. **1**). capC-MAP outputs are in the "bedGraph" file format which can be read by many downstream analysis and visualisation tools. Full details are given in the capC-MAP user manual.

To our knowledge capC-MAP is the only software specific to the Capture-C experimental design where an interaction profile is obtained for each targeted restriction enzyme fragment. It is possible to instead use software designed to treat HiC data, and then perform additional Capture-C specific analysis steps manually. One tool which has facilities to do this is HiC-Pro; in Supplementary Information we compare the performance of HiC-Pro and capC-MAP. For testing we used a data set with 35 targets captured from mouse erythoid cells (obtained from Davies *et al.* (2016)), and a data set with 446 targets captured from developing mouse midbrain cells (obtained from Andrey *et al.* (2017)). capC-MAP identifies a higher proportion of PCR duplicates, finds on average about twice as many informative reads, and performs the analysis in under a quarter of the time taken by HiC-Pro. This highlights the fact that the packages are optimized for different types of data. Full details are given in Supplementary Information and Supplementary Table 1. Another common method, which is similar but distinct from Capture-C, is "Capture HiC"; there, different experimental designs are common, and other software may be more appropriate (see Supplementary Information).

capC-MAP is freely available under the GNU General Public License v3.0, and can be obtained from https://github.com/cbrackley/capC-MAP, with user manual at capc-map.readthedocs.io. It can also be installed via the conda package manager. capC-MAP comes with a small example data set, and several "worksheets" showing examples of how plots such as Fig. **1** can be generated using different tools (either R packages or command-line based tools common in NGS bioinformatics). In Supplementary Information we give full details of the processing steps performed by the software, as well as background details on the Capture-C method.

Funding

This work was supported by the European Research Council [grant no. 648050, THREEDCELLPHYSICS], and the UK Medical Research Council [grant no. MR/J00913X/1].

References

Andrey,G. *et al.* (2017). Characterization of hundreds of regulatory landscapes in developing limbs reveals two regimes of chromatin folding. *Genome Res.*, **27**, 223–233.

Buckle,A., *et al.* (2018). Polymer simulations of heteromorphic chromatin predict the 3-d folding of complex genomic loci. *Mol. Cell*, **72**, 786–797.

Davies,J. *et al.* (2016). Multiplexed analysis of chromosome conformation at vastly improved sensitivity. *Nat. Methods*, **13**, 74–80.

Dekker,J. *et al.* (2002). Capturing chromosome conformation. *Science*, **295**, 1306–1311.

Denker, A. and de Laat, W. (2016). The second decade of 3c technologies: detailed insights into nuclear organization. *Gene Dev.*, **30**, 1357–1382.

Furlan,G. *et al.* (2018). The ftx noncoding locus controls x chromosome inactivation independently of its rna products. *Mol. Cell*, **70**, 462–472.e8.

Han,J. *et al.* (2018). 3c and 3c-based techniques: the powerful tools for spatial genome organization deciphering. *Mol. Cytogenet.*, **11**, 21.

Hughes,J.R. *et al.* (2014). Analysis of hundreds of cis-regulatory landscapes at high resolution in a single, high-throughput experiment. *Nat. Genet.*, **46**, 205–212.

Langmead,B. *et al.* (2009). Ultrafast and memory-efficient alignment of short dna sequences to the human genome. *Genome Biology*, **10**, R25.

Li,H., *et al.* (2009). The sequence alignment/map format and samtools. *Bioinformatics*, **25**, 2078–2079.

Lieberman-Aiden,E. *et al.* (2009). Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science*, **326**, 289–293.

Martin,M. (2011). Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet.journal*, **17**.

Simonis,M. *et al.* (2006). Nuclear organization of active and inactive chromatin domains uncovered by chromosome conformation capture-on-chip (4c). *Nat. Genet.*, **38**, 1348.

Xia,Q. *et al.* (2016). The type 2 diabetes presumed causal variant within tcf7l2 resides in an element that controls the expression of acsl5. *Diabetologia*, **59**, 2360–2368.

capC-MAP : software for analysis of Capture-C data

Adam Buckle[2], Nick Gilbert[2] Davide Marenduzzo[1], and Chris A. Brackley[1*]

1 SUPA, School of Physics and Astronomy, University of Edinburgh, Peter Guthrie Tait Road, Edinburgh EH9 3FD, United Kingdom

2 Medical Research Council Human Genetics Unit, Medical Research Council Institute of Genetics & Molecular Medicine, University of Edinburgh, Western General Hospital, Edinburgh, United Kingdom

*To whom correspondence should be addressed.

Supplementary Information

Here we give full details of the analysis pipe-line which the capC-MAP software performs. For clarity we begin with a summary of the Capture-C method, introducing some key terms. Finally, we compare capC-MAP with some other software tools, providing a comparison of performance against HiC-Pro, a tool for analysing HiC data which also has options for Capture-C data.

The Capture-C method

The Capture-C protocol is described in Refs. [1, 2]; for completeness here we summarize the main details of a typical experiment and introduce some terminology.

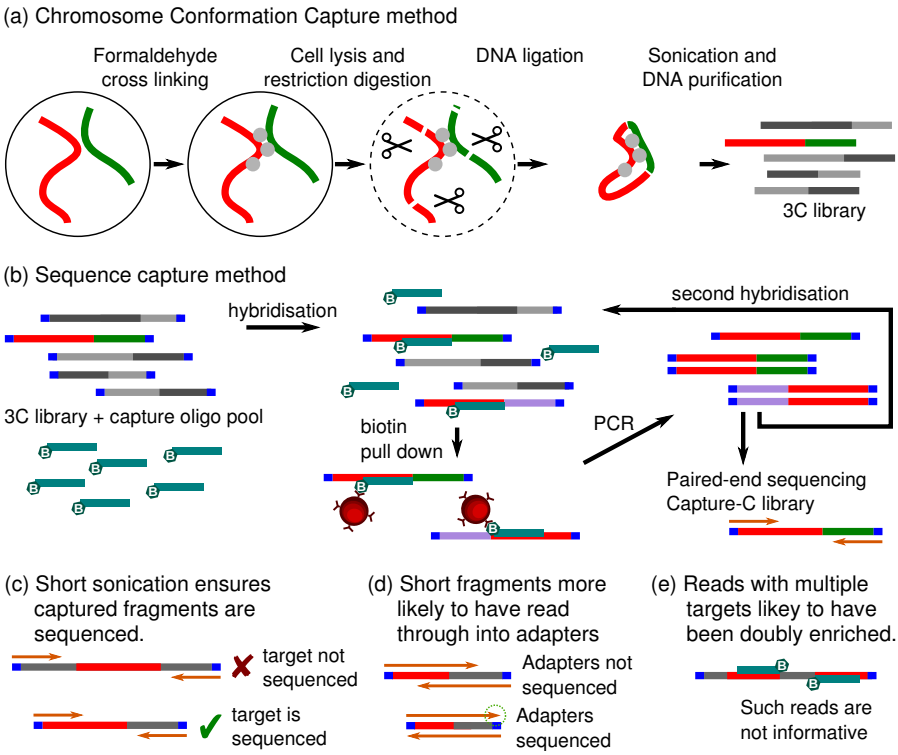
The underlying principles of all chromosome-conformation-capture (3C) based methods are similar and summarised in Suppl. Fig. S1a. First, formaldehyde fixation is used to cross-link proteins and DNA within intact nuclei, physically linking DNA segments which are in close spatial proximity. Next, the formaldehyde cross-linked DNA template is digested in the intact nuclei [3] with a selected restriction enzyme, and then the DNA is re-ligated. Since ligation is likely to occur between cross-linked fragments, this results in the joining of fragments that were not adjacent in the linear genome, but were close together in 3-D space. The resulting DNA is purified to form a 3C library. In the Capture-C method the 3C library is then sonicated to an optimal size of ~ 300 bp, and used to prepare sequencing libraries, whereupon solution-based sequence capture technology is used to enrich for certain restriction enzyme fragments. Specifically, biotin labelled RNA or DNA capture oligos are designed against a set of restriction fragments of interest – these hybridise with the DNA fragments, which are then pulled down with the biotin tag, before re-amplification using primers to sequencing adapters. Since the library consists of hybrid fragments representing the proximity ligation events, paired-end sequencing reveals which distal fragments were in proximity to the fragments of interest (Suppl. Fig. S1b). Here, we use the terminology "targets" to refer to the restriction enzyme fragments for which oligos have been designed, and "reporters" to refer to any fragments which have been found ligated to a target. The set of reporters for a given target can be used to build up a picture of the interactions genome wide (they are "piled-up" to provide an "interaction profile"). The data generated from Capture-C is similar to 4C, but here the capture oligos provide the viewpoints, so multiple interaction profiles can be obtained from a single experiment (note that our term *target* is synonymous with the *viewpoint* or *bait* in 4C).

Capture-C uses a restriction endonuclease with a four base-pair recognition sequence (typically *Dpn*II); the short recognition sequence means it appears frequently within the genome, resulting in short restriction fragments. This – together with the oligo hybridisation step which vastly improves the signal to noise ratio by reducing the number of background ligation events being sequenced – leads to very high-resolution data. The short restriction fragment size necessitates that the library be sonicated to a similar short length (compared to that typical of HiC experiments) to ensure that the captured fragment falls within the sequenced region (see Suppl. Fig. S1c).

There are several possible oligo design strategies, but typically this entails designing oligos which bind each end of the target fragments. In the original Capture-C method [1] Hughes *et al.* used RNA oligos synthesized on a microarray (meaning that the design included a minimum of 40,000 oligos), designed such that each end of each target fragment was tiled by several oligos. In a modified version of the method (named Next Generation (NG) Capture-C [2]) a single 120 bp biotinylated DNA oligo was designed for each end of every target fragment – this allowed for a more cost effective and scalable experimental design. The Hughes lab developed an on-line oligo design tool called CapSequm (http://apps.molbiol.ox.ac.uk/CaptureC/cgi-bin/CapSequm.cgi) which performs a BLAT search [4] and Repeatmasker analysis [5] to generate robust capture oligos which will hybridise to a single restriction fragment. Another improvement in the NG Capture-C protocol is that two successive rounds of sequence capture are performed, the first giving a 5-20,000 fold enrichment, with the second able to achieve up to 1,000,000-fold enrichment. This dramatically improves the signal-to-noise ratio and reduces the required sequencing depth. This efficiency also offers the ability to pool multiple 3C libraries with indexed sequencing adapters, which can then be processed in a single reaction [2].

capC-MAP overview

In developing capC-MAP our aim was to automate the analysis of Capture-C data, going from fastq files of sequenced reads to a set of outputs for each target using a single command line. The main output from the software is an "interaction profile" for each target, showing intrachromosomal interactions (interchromosomal interactions are output separately). Using an easily customisable "configuration" file the user can specify different normalization and binning options. Interaction profiles are output in the stan-



Suppl. Fig. S1: Schematic showing the steps of the Capture-C method. (a) 3C library preparation steps: formaldehyde cross-linking; cell lysis; digestion of cross-linked chromatin by the *Dpn*II restriction endonuclease; ligation by T4 DNA ligase, joining hybrid DNA fragments together; and finally, DNA sonication and purification to produce a 3C library. **(b)** Sequence capture steps of the Capture-C methodology: Illumina sequencing adapters are added (blue); an experiment specific biotin labelled capture oligo pool is combined with the 3C library in a hybridisation reaction; streptavidin beads (red circles) are used to pull down biotin oligo/3C DNA complexes; and then the library is re-amplified using primers to the Illumina sequencing adapters. After this, a second round of hybridisation and amplification can be performed before paired-end sequencing. Figure adapted from Refs. [1, 2]. **(c-e)** Some technical points of the Capture-C method. Sonication to a small size (200-300 bp), similar to the length of *Dpn*II restriction fragments) is recommended to ensure that the captured targets are sequenced. Paired-end sequencing of short fragments is likely to lead to adapter contamination as a result of read through into $3'$ end. Ligation fragments with multiple targets may have been captured multiply; since oligo efficiency is generally unknown, such fragments are not quantitatively informative.

dard bedGraph format – there are many tools available for visualization and downstream analysis of data in this format. For example, IGV [7] or the UCSC genome browser [8] can be used for visualization, and the BEDtools suit [9] or many of the R packages available via bioconductor [10] can be used for downstream analysis and plotting (for example the "peakC"package performs non-parametric peak calling on 4C and Capture-C data [11]). The capC-MAP documentation includes example commands and scripts for using these tools to treat capC-MAP output.

capC-MAP details

The analysis pipe-line which capC-MAP follows is based on that detailed in Ref. [1], and is shown schematically in Suppl. Fig. S2. Here, we summarise the steps.

- Since it is recommended that during library preparation fragments are sonicated to an average length of 200-300 bp, it is likely that read through into the adapter sequence will have occurred during paired-end sequencing (see Suppl. Fig. S1d). The first step in the analysis is therefore to trim adapter sequence from the mapped reads – this is done using the **cutadapt** software [12] (trimming the common adapter sequence).

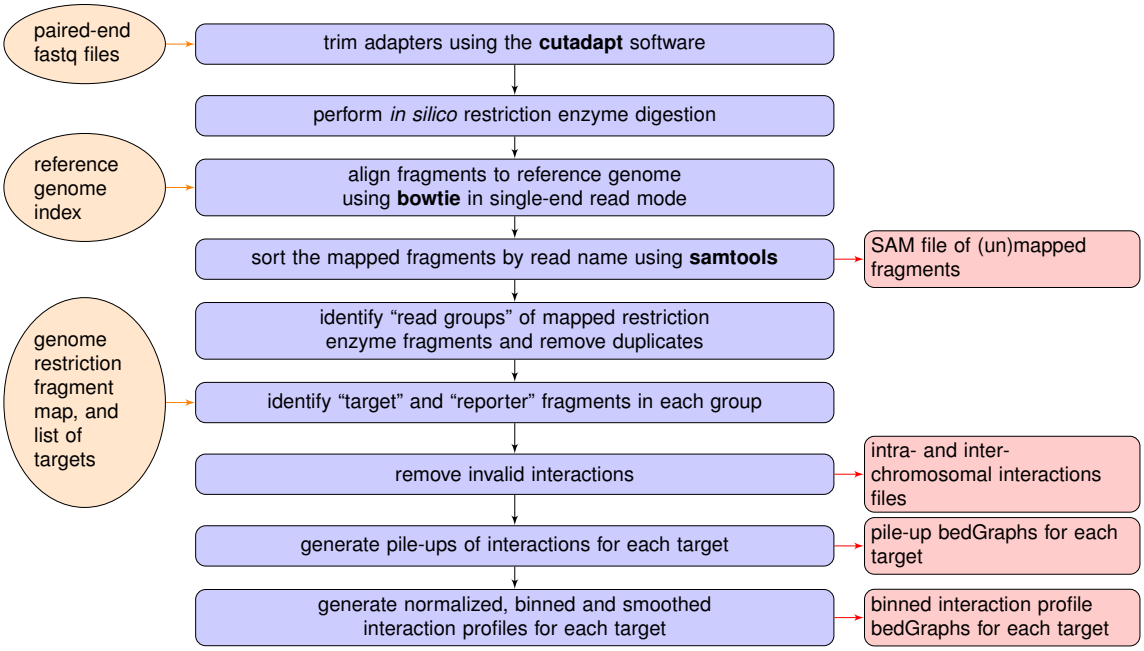
- Next, we perform an *in silico* restriction enzyme digestion; i.e. each read-pair is searched for instances of the enzyme cut sequence (GATC for *Dpn*II), and is broken into smaller fragments at these positions. (These sequences will be found at ligation junctions between restriction enzyme fragments, so the sequence either side must be mapped independently to the reference genome.) Thus, a group of read fragments is obtained from the read pair.

- The group of fragments is then aligned to the reference genome using the **bowtie** software [6], as though they were single-end reads (Bowtie would make incorrect assumptions about valid alignments if run in paired-end mode). Since the fragments may be quite short, Bowtie is run with quite stringent reporting criteria to ensure only uniquely mapped fragments are reported. This is the most time consuming step of the analysis, and can be run in parallel on a multi-core computer.

- The output from Bowtie is a SAM format file containing details of all mapped (and unmapped) fragments. This needs to be sorted by read name so that the original fragment groups can be recovered. At this point duplicates are removed: these are defined as read groups where identical fragments appear in the same order, and two fragments are said to be identical if either they mapped to the same position in the genome, or they did not map but have identical sequence. Such duplicates are likely to have arisen from PCR artefacts.

- Next, the set of mapped fragments within each read group is compared to a genome wide map of restriction enzyme fragments and to the list of target fragments, in order to identify "targets" and "reporters".

- At this point, invalid interactions are identified and removed, and the remaining valid intra- and interchromosomal interactions are stored separately for each target. Invalid interactions are
  - interactions between targets; since these will have resulted from ligation events which bring together regions which



Suppl. Fig. S2: Flow diagram for the Capture-C analysis process. Each of the steps completed by the capC-MAP software during a typical run is shown (centre). Where external software is called this is shown in bold. Required inputs are shown to the left (the reference genome index is generated using the Bowtie alignment software [6], and the genome-wide map of restriction enzyme fragments can be generated by capC-MAP). Typical outputs are shown to the right.

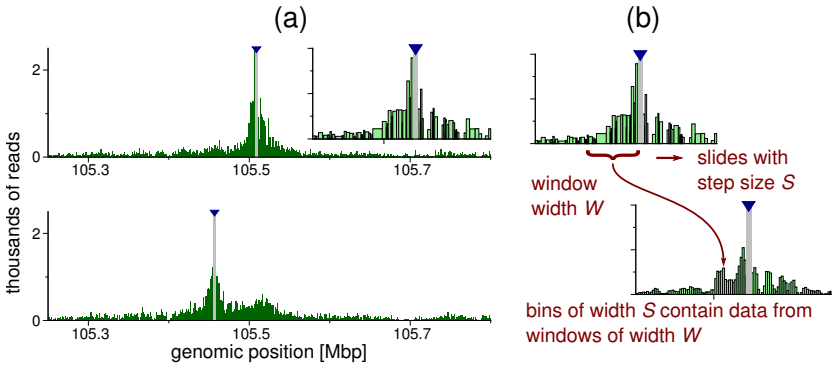
bind more than one oligo they are likely to have been doubly enriched and are therefore not quantitative (Suppl. Fig. S1e).
  - interactions with an "exclusion zone" around another target; since digestion is not 100% efficient, these could also have resulted from ligation events which bring together regions which bind more than one oligo.
  - interactions with multiple (non-adjacent) reporters; it is possible to find fragments which map to more than two distal chromatin regions within the same read group – though these are in theory informative [13, 14], in practice they occur very rarely, so for simplicity we treat them as invalid.
- Finally, for each target the list of intrachromosomal interactions is "piled-up" to generate a bedGraph format file which counts the number of interactions between that target and every other restriction enzyme fragment in the same chromosome. This gives a raw restriction enzyme fragment level interaction profile. Depending on the read coverage it can also be useful to generate a binned and smoothed interaction profiles (detailed below).

capC-MAP outputs a set of informative log files and intermediate data files as detailed in the documentation, but the main output is an interaction profile for each target in the standard bedGraph file format. This is obtained by piling-up interactions for each reporter restriction enzyme fragment; a typical plot of this data is shown in Suppl. Fig. S3a (since restriction enzyme fragments have different sizes, the bars in this plot are of different widths). capC-MAP can also perform normalization of each profile to remove biases which arise because different oligos have different capture efficiencies (we use the assumption that each target should show equal visibility in its interactions genome-wide, and interactions are reported in units of "reads per million").

It is often useful to apply some binning or smoothing to interaction profiles, and capC-MAP provides options to do this. For example, if a data set has a low read count per target, binning can give smoother and easier to interpret interaction profiles. Also, since the length of restriction fragments has quite a broad distribution, examining raw pile-ups can be misleading; e.g, if we consider two regions which interact with a target at a similar frequency – if one region has a single long restriction enzyme fragment, and the other has several short ones, in the latter case the same number of interactions would be shared across more fragments, resulting in a lower 'per fragment' interaction count. capC-MAP uses sliding window binning, where the user specifies a window width W and a step size S, where $W \geq S$. Bins go up in steps of size S bp, and each contains reads from a window of width W bp around the bin centre (this strategy is shown schematically in Suppl. Fig.S3b). It can be informative to generate profiles with several different bin/window size combinations, e.g. depending on whether short or longer ranged interactions are of interest; Fig. 1 in the main text shows examples of different binned and smoothed interaction profiles.

Implementation

capC-MAP is implemented as a suit of programs written in C++ and Python. Each program can be run separately, or an entire analysis can be performed with a single command line. capC-MAP also calls the external software packages cutadapt [12], Bowtie [6] and samtools [16], which are freely available open-source software. capC-MAP is available under the GNU General Public License v3.0 licence, and can be obtained from https://github.com/cbrackley/capC-MAP, with full documentation at capc-map.readthedocs.io. It should run on any standard Unix-style system (including Linux and Mac) where the above



Suppl. Fig. S3: Typical interaction profiles and the smoothing and binning scheme. (a) Plots showing raw pile-ups of reads for two different targets from experimental data published in Ref. [15] (these are targets on chromosome 2 of mouse mm9 build; data obtained from GEO:GSE120666). The grey bar and blue arrowhead indicate the position of the target. Inset shows a zoom around the target shown in the top plot – here it is evident that restriction fragments are of different sizes. **(b)** Schematic showing the sliding window binning scheme used by capC-MAP. Note that capC-MAP does not provide functionality to generate plots since there are many existing software tools which can read and plot bedGraph files.

listed software is installed. We have also made capC-MAP available via the bioconda channel [17] for the conda package manager which allows installation of all requirement with a single command.

For a typical Capture-C experiment, the user will need to perform the following steps:

1. Build an index for the reference genome for the Bowtie alignment software;
2. Build a restriction enzyme fragment map for the reference genome using the capC-MAP "genomedigest" tool;
3. Run the full analysis pipe line using the capC-MAP "run" tool;

where steps 1 and 2 only need to be performed once for each reference genome. Bowtie indexes can be downloaded pre-built, though it is essential that these are built from the same reference genome from which the restriction enzyme fragment map is built in step 2. Step 3 requires only a single command line, and the software reads a "configuration file" to set all the options. A template configuration file is provided with the software, and this is straightforward to modify for a specific experiment. capC-MAP can run some steps in parallel by taking advantage of the shared memory multi-threading options of the external programs it calls. capC-MAP also provides functionality for handling targets which appear at multiple points in a genome, and for combining replicate experiments. capC-MAP comes with a small example data set and several "worksheets" showing examples of how plots such as presented in Fig. 1 in the main text can be generated using different tools (either R packages or command-line based tools common in NGS bioinformatics).

Comparison with other software

To our knowledge, the only other publicly available software which automates analysis of Capture-C data is HiC-Pro [18]. This is a popular tool for HiC data analysis, and a recent update added the ability to analyse Capture-C data. Other tools which could possibly be used to treat Capture-C data include HiCUP [20] and HOMER [21], but neither of those software packages can extract individual interaction profiles, meaning the user would have to write custom scripts to perform the final steps of the analysis – for that reason we do not compare them directly with capC-MAP here.

In order to compare the efficiency of HiC-Pro and capC-MAP we used two different, publicly available data sets. Data set A was obtained from Ref. [2], where oligos were designed for 35 targets across the mouse genome, and interactions captured from mouse erythoid cells. Data set B was obtained from Ref. [19], where oligos for 446 targets, again in the mouse genome, were used. That work aimed to study differences in chromatin interactions in different embryonic tissues at different time-points during development; the specific data set we used was from midbrain in day 10.5 embryos. In both cases two processor cores were used, and all standard options selected (HiC-Pro was run in 'sequential mode' where some processing steps which are only relevant for HiC data were skipped). Some details of the analysis from each software package are shown in Table 1; note that the packages may count reads in different ways meaning that not all quantities are directly comparable. We find that capC-MAP identifies a higher proportion of PCR duplicates, and finds on average about 2.1 times more informative reads; also capC-MAP performs the analysis in under a quarter of the time taken by HiC-Pro on average, highlighting that the packages are optimized for different types of data. A major difference between the two pipelines is that capC-MAP performs an *in silico* digestion of the reads before alignment of the resulting fragments, whereas HiC-Pro attempts alignment before digestion and only searches for enzyme cut sites if this fails (i.e., where a ligation junction appears within the read, alignment is attempted multiple times). The latter strategy is optimal for HiC data where typically a less frequently cutting enzyme is used (e.g. *Hind*III) and the ligation fragments tend to be longer, meaning the sequenced regions are less likely to contain a cut site; in the case of Capture-C using *Dpn*II, the fragments are highly likely to include a cut site, so for most fragments HiC-Pro will need to attempt alignment multiple times. Another possible reason for the lower efficiency of HiC-Pro is that it uses the Bowtie2 aligner [22], whereas capC-MAP uses Bowtie1 which is optimised for short reads [6].

As well as being optimized to analyse Capture-C data more efficiently, capC-MAP also has additional features which are useful for downstream analysis. First, capC-MAP has options for interaction profile smoothing: this is particularly useful for data with low read coverage per target, where simple binning would result in a noisy profile which might be difficult to interpret. An example is shown in Fig. 1 in the main text: the same interaction data is plotted with different binning and smoothing options, showing that different features can be observed at different scales. As detailed above, capC-MAP normalizes the data from each

Data set A details	
Total reads	37,381,686
Number of target fragments	35

Data set B details	
Total reads	89,501,599
Number of target fragments	446

	Data set A		**Data set B**	
	capC-MAP	**HiC-Pro**	**capC-MAP**	**HiC-Pro**
number of duplicates removed	3,391,919 (9.07 %)	485,560 (1.29 %)	29,866,200 (33.37 %)	7,912,872 (8.8 %)
reads without target-reporter pair	32,659,794	–	57,390,395	–
number invalid interactions removed	172,596	–	354,666	–
total informative reads	1,140,683	667,259	1,859,095	723,417
of which were interchromosomal	290,516	106,348	378,033	136,976
of which were intrachromosomal	850,167	560,911	1,481,062	586,441
average interchromosomal per target	8,300	3,038	847	307
average intrachromosomal per target	24,290	16,026	3,320	1,315
total run time	3 hours 31 mins	11 hours 39 mins	10 hours 28 mins	84 hours 16 mins

Suppl. Table 1: Table comparing output from capC-MAP v0.0.1 and HiC-Pro v2.11 [18]. Data sets from (A) Ref. [2] (GEO:GSE67959) and (B) Ref. [19] (GEO:GSM2251422) were used to compare the two software packages, run using two cores of a 10 core hyper-threaded 2.6GHz Intel Xeon E5-2660 processor machine with 50GB RAM running Scientific Linux 7.5. Note that the two software packages report read statistics in different ways, so may not be directly comparable, and not all values are available from both. (Note that times given for HiC-Pro include two steps: first the pipeline is run to obtain genome-wide interactions, before the provided script is used to extract each target separately. The time taken for this second step grows with the number of targets and, for example, accounts for ∼68 hours of the total time take to analyse data set B.)

target independently to obtain "reads-per-million" profiles, and also reports per-target interaction statistics (interchromosomal *vs.* intra chromosomal interactions, local *vs.* long-rage interactions etc.). We note that users of HiC-Pro (or other tools designed with HiC data in mind) would have to write custom scripts to correctly normalize and generate these binned profiles. capC-MAP also allows replicate data to be combined using a single command: in a typical work-flow, data from each replicate would be analysed separately to ensure similar interactions were obtained, then replicates can be combined to obtain profiles at greater read coverage. Finally, as part of the documentation for capC-MAP, we provide a set of example R scripts (which use common bioconductor libraries) and example Python and BEDtools [9] commands, which will allow the user to quickly produce plots such as those shown in Fig. 1 in the main text.

capC-MAP was designed with the intention of extracting interaction profiles (such as might be obtained in a 4C experiment) for each targeted restriction enzyme fragment. Other experimental designs include (i) capture oligos designed to tile a region or chromosome of interest to obtain a HiC-style map (as in Ref. [23]); and (ii) oligos designed to capture interactions from many thousands of dispersed sites (e.g. all promoters), to identify significant interactions between target and non-target or between pairs of target sites (as in Refs. [24, 25]). These approaches often use the "Capture Hi-C" protocol [24, 26, 27] which combines elements from the Capture-C and Hi-C methods. For such experiments, analysis strategies different from those employed by capC-MAP might be more relevant, and tools such as HiC-Pro [18] and CHiCAGO [27] (which is designed specifically for experiments of type (ii)) may be more appropriate. We note that in design case (ii) it is still possible to use capC-MAP to generate interaction profiles for each target, though their quality will depend on the read depth (which may be lower than a Capture-C experiment if the same number of reads are diluted across thousands of targets).

References

[1] J. R. Hughes, et al. *Analysis of hundreds of cis-regulatory landscapes at high resolution in a single, high-throughput experiment*. Nature Genetics **46**, 205 (2014).

[2] J. Davies, et al. *Multiplexed analysis of chromosome conformation at vastly improved sensitivity*. Nature Methods **13**, 74 (2016).

[3] A. Gavrilov, et al. *Disclosure of a structural milieu for the proximity ligation reveals the elusive nature of an active chromatin hub*. Nucleic Acids Research **41**, 3563 (2013).

[4] K. WJ. *BLAT–the BLAST-like alignment tool*. Genome Research **12**, 656 (2002).

[5] A. Smit, R. Hubley, and P. Green. *RepeatMasker Open-4.0*. http://www.repeatmasker.org (2013-2015).

[6] B. Langmead, et al. *Ultrafast and memory-efficient alignment of short DNA sequences to the human genome*. Genome Biology **10**, R25 (2009).

[7] H. Thorvaldsdóttir, J. T. Robinson, and J. P. Mesirov. *Integrative Genomics Viewer (IGV): high-performance genomics data visualization and exploration*. Briefings in Bioinformatics **14**, 178 (2013).

[8] W. J. Kent, et al. *The Human Genome Browser at UCSC*. Genome Research **12**, 996 (2002).

[9] A. R. Quinlan and I. M. Hall. *BEDTools: a flexible suite of utilities for comparing genomic features*. Bioinformatics **26**, 841 (2010).

[10] W. Huber, et al. *Orchestrating high-throughput genomic analysis with Bioconductor*. Nature Methods **12**, 115 (2015).

[11] G. Geeven, et al. *peakC: a flexible, non-parametric peak calling package for 4C and Capture-C data*. Nucleic Acids Research **46**, e19 (2018).

[12] M. Martin. *Cutadapt removes adapter sequences from high-throughput sequencing reads*. EMBnet.journal **17** (2011).

[13] P. Olivares-Chauvet, et al. *Capturing pairwise and multi-way chromosomal conformations using chromosomal walks*. Nature **540**, 296 (2016).

[14] A. M. Oudelaar, et al. *Single-cell chromatin interactions reveal regulatory hubs in dynamic compartmentalized domains*. bioRxiv 307405; doi:10.1101/307405 (2018).

[15] A. Buckle, et al. *Polymer Simulations of Heteromorphic Chromatin Predict the 3-D Folding of Complex Genomic Loci*. Molecular Cell **72**, 786 (2018).

[16] H. Li, et al. *The Sequence Alignment/Map format and SAMtools*. Bioinformatics **25**, 2078 (2009).

[17] B. Grüning, et al. *Bioconda: sustainable and comprehensive software distribution for the life sciences*. Nature Methods **15**, 475 (2018).

[18] N. Servant, et al. *HiC-Pro: an optimized and flexible pipeline for Hi-C data processing*. Genome Biology **16**, 259 (2015).

[19] G. Andrey, et al. *Characterization of hundreds of regulatory landscapes in developing limbs reveals two regimes of chromatin folding*. Genome Research **27**, 223 (2017).

[20] S. Wingett, et al. *HiCUP: pipeline for mapping and processing Hi-C data*. F1000Research **4**, 1310 (2015).

[21] S. Heinz, et al. *Simple combinations of lineage-determining transcription factors prime cis-regulatory elements required for macrophage and B cell identities*. Molecular cell **38**, 576 (2010).

[22] B. Langmead and S. L. Salzberg. *Fast gapped-read alignment with Bowtie 2*. Nature Methods **9**, 357 (2012).

[23] A. L. Sanborn, et al. *Chromatin extrusion explains key features of loop and domain formation in wild-type and engineered genomes*. Proceedings of the National Academy of Sciences USA **112**, E6456 (2015).

[24] B. Mifsud, et al. *Mapping long-range promoter contacts in human cells with high-resolution capture Hi-C*. Nature Genetics **47**, 598 (2015).

[25] A. Chesi, et al. *Genome-scale Capture C promoter interaction analysis implicates novel effector genes at GWAS loci for bone mineral density*. bioRxiv 405142; doi:10.1101/405142 (2018).

[26] S. Schoenfelder, et al. *The pluripotent regulatory circuitry connecting promoters to their long-range interacting elements*. Genome Research **25**, 582 (2015).

[27] J. Cairns, et al. *CHiCAGO: robust detection of DNA looping interactions in Capture Hi-C data*. Genome Biology **17**, 127 (2016).